Expert Systems with Applications 36 (2009) 12044–12048

Contents lists available at ScienceDirect

Expert Systems with Applications

journal homepage: www.elsevier.com/locate/eswa

# Evaluation of fuzzy neural network run-to-run controller using numerical simulation analysis for SISO process

Chien-Chih Wang [a,*], Bernard C. Jiang [b], Ming-Yih Wu [b], Chih-Hung Jen [c]

[a] Department of Industrial Engineering and Management, Ming Chi University of Technology, Taipei County, Taiwan
[b] Department of Industrial Engineering and Management, Yuan Ze University, Taiwan
[c] Department of Information Management, Lung Hwa University of Science and Technology, Taiwan

ARTICLE INFO

Keywords:
FNN
EWMA
Adaptive EWMA
Large autoregressive

ABSTRACT

During the past decade, a variety of run-to-run (R2R) control techniques have been proposed and extensively used to control various semiconductor manufacturing processes. The R2R control methodology combines response surface modeling, engineering process control, and statistical process control, with the main objective of fine-tuning the recipe so that the process output of each run can be maintained as close to the nominal target as possible. In this paper, the single-input single-output (SISO) model is addressed. To overcome the shortcomings in the traditional R2R EWMA controller, a fuzzy neural network (FNN) control strategy is proposed. When a process has large autoregressive parameters, traditional EWMA control methods cannot establish stable SISO process control. To solve this problem, an SISO process control model based on an FNN was used to build an SISO process control procedure. The analysis results from a numerical simulation indicated that when the coefficient of autocorrelation $\phi > 0.6$, the MSE ratio when using the FNN controller was 97.11% lower than when using the EWMA controller and 61.12% lower than when using an adaptive EWMA controller. This showed that the FNN control method established better SISO process control than the EWMA and adaptive EWMA control methods.



## 1. Introduction

Semiconductor manufacturing is arguably the fastest evolving industry in the world. However, success in this industry requires constant attention to state of the art process tools, processing techniques, and process improvement. In a typical semiconductor manufacturing system, within-run (or batch) variation is usually controlled by automatic controllers built into the equipment. A run-to-run (R2R) controller is necessary because the equipment will experience aging and/or wear over time. Maintenance operations can also change the operating conditions for a process. Therefore, an R2R controller is needed to act as a supervisor to indicate whether a recipe change is needed and suggest new recipes for use in the next product batch (Adivikolanu & Zafiriou, 2000; Butler & Stefani, 1994; Del Castillo & Hurwitz, 1997; Patel & Jenkins, 2000; Sachs, Hu, & Ingolfsson, 1991, 1995).

Many researchers have discussed R2R control in the semiconductor industry. Ingolfsson and Sachs (1993) and Sachs et al. (1995) developed a preliminary R2R control scheme that used a simple exponentially weighted moving average (EWMA) R2R controller statistic to obtain an appropriate recipe adjustment for a silicon epitaxial growth process in a barrel reactor. Del Castillo (2002) discussed the chemical mechanical planarization (CMP) process, using SISO process model response values for the removal rate and to control the variable platen speed. With their method the coefficient of autocorrelation may approach 0.9006. For this reason, the dynamic situation between the current and next batch may exhibit high practical process autocorrelation levels. Processes with large autoregressive parameters, as discussed in Jen, Jiang, and Fan (2004) cannot be well controlled using traditional EWMA control methods to control a single-input single-output (SISO) process. To overcome this problem, this paper proposes a fuzzy neural network (FNN) approach, in which fuzzy logic is integrated with a network's learning ability. In the past, FNNs have been directly incorporated with the known behavioral aspects of a process, leaving the neural network to empirically capture the unknown aspects. Fuzzy inference systems have been applied in process modeling, adaptive control, and model predictive control (Hasegawa, Horikawa, Furuhasha, & Uchikawa, 1995; Lee & Park, 1992; Lin & Lee, 1994; Shen, Cao, Zhu, & Sun, 2002). According to these studies, an FNN controller is one of the more promising approaches in the R2R control field.

Due the SISO R2R controller can compensate for most of the process dynamics and noise disturbances, avoiding complicated calculations. The focus of this paper is an SISO process system with a large autoregressive parameter and the FNN controller is proposed and analyzed. This paper is outlined as follows. In the next

* Corresponding author. Tel.: +886 2 29089899x4713.
E-mail address: ieccwang@mail.mcut.edu.tw (C.-C. Wang).

0957-4174/$ - see front matter © 2009 Elsevier Ltd. All rights reserved.
doi:10.1016/j.eswa.2009.03.011

section, the structure and procedure of the FNN R2R controller will be introduced. In Section 3, a practical SISO model is simulated, and the control performance of the FNN controller is compared with other controllers. Section 4 concludes the paper and summarizes our findings.

## 2. The proposed FNN SISO R2R controller

The proposed FNN R2R controller was designed to use two current outputs and the next measured run as the input to the FNN SISO process system. These outputs are treated as the input variables for the controller. Simultaneously, the control variable ($\mu_t$) of the current period ($t$) is taken as the FNN controller output variable. Since this FNN controller design was implemented to forecast controllable variables, it can also consider time-varying situations. The controller applies the difference between the nominal target and actual response in each run to identify and adjust the control error, making the output more accurate to target. The FNN controller's parameter weights are regulated based on the fuzzy technique with a back-propagation (BP) learning rule to obtain better performance.

In this SISO R2R controller, both the response and controllable variables for the current period affect the next run. Therefore, the FNN procedure responds to the current period response and the expected response for the next period to infer the forecasting variable ($\hat{\mu}_t$). The controllable variable ($\hat{\mu}_t$) is used to replace the process model variable from the previous period to carry out control. The measured output is used to obtain output for the next run ($y_{t+1}$). In this paper, a more generalized SISO process model is adopted (Del Castillo, 2002; Sachs, Hu, & Ingolfsson, 1995) and the practical response from the next run can be expressed by

$$y_{t+1} = \phi y_t + \alpha + \beta\hat{\mu}_t + N_{t+1}. \tag{1}$$

In Eq. (1), $y_{t+1}$ is the quality characteristic, $\phi$ is the autocorrelation coefficient, and $\alpha$ and $\beta$ are the process offset and first-order term of the control variable $\hat{\mu}_t$, respectively. $N_{t+1}$ is the colored noise term in the ARIMA model form. If $N_{t+1}$ follows ARIMA (1, 0, 1), then $N_{t+1} = (1 - cz^{-1}/1 - \omega z^{-1})\varepsilon_{t+1}$, where $c$ and $\omega$ are the first-order MA and AR coefficients, respectively. Again, the noise term $\varepsilon_{t+1}$ is assumed to be white, i.e. $\varepsilon_{t+1} \overset{i.i.d}{\sim} N(0, \sigma^2)$.

Because the expected response can reach the nominal target ($T$), Eq. (1) becomes the following:

$$T = \phi y_t + \alpha + \beta\mu_t + N_{t+1}. \tag{2}$$

In Eq. (2), $\mu_t$ is assumed to be a virtual value. Furthermore, the response for the next run can be controlled as close to the nominal target as possible via the controllable variable ($\mu_t$). If Eq. (1) is subtracted from Eq. (2), then the control law can be derived to be

$$\mu_t - \hat{\mu}_t = \frac{T - y_{t+1}}{\beta}, \tag{3}$$

where $\beta$ represents the parameter being estimated applied to the linear model using the regression method. From Eq. (3), the error value between the nominal target ($T$) and the response for the next period ($y_{t+1}$) can be successfully transformed to obtain the expected controllable variable.

The input–output relation of the proposed FNN is defined by the extension principle (Lee & Teng, 2000). The FNN model has four layers, including the input layer, membership layer, rule layer, and output layer. In the FNN model, the node weights $W_{ij}$, $m_{ij}$, $\sigma_{ij}$ are adjusted to implement the back-propagation and fuzzy learning rule.

The output layer is adjusted using parameter $W_{ij}$. It can be derived as follows:

$$-\frac{\partial E}{\partial W_{ij}^4} = \frac{T - y_{t+1}}{\beta} x_i^4, \tag{4}$$

where $x_i^4$ is the input value from the $i$th node of the fourth layer. The error function ($E$) can be represented as

$$E = \sum_{i=1}^{p} \frac{1}{2}(d_i(t) - y_i(t))^2 = \sum_{i=1}^{p} \frac{1}{2}\left(\frac{T - y_{t+1}}{\beta}\right)^2. \tag{5}$$

In Eq. (5), $p$ is the number of output nodes (in this case $p = 1$), $y_i(t)$ is the actual output of period $t$, and $d_i(t)$ is the expected output. The weight of the nodes for the next period is as follows:

$$W_{ij}^4(t+1) = W_{ij}^4(t) + \eta_W \cdot \delta_j^4(t) \cdot x_i^4(t). \tag{6}$$

In the above equation, $\delta_j^4 = (\mu_t - \hat{\mu}_t) = (T - y_{t+1})/\beta$, and $\eta_W$ is the learning rate of $W_{ij}^4$. The rule layer is designed to calculate the error of the returned pass $\delta_i^3$ and can be derived as follows:

$$\delta_i^3 = -\frac{\partial E}{\partial \mu_i^3} = \sum_{j=1}^{p} \delta_j^4 \cdot W_{ij}^4. \tag{7}$$

The membership layer adopts a Gaussian function as the membership function. In this layer, the adjusted parameters are $m_{ij}$ and $\sigma_{ij}$, identified as the mean and standard deviation of the Gaussian function, respectively. According to Eq. (4), the regulated rule of $m_{ij}$ can be derived as

$$-\frac{\partial E}{\partial m_{ij}} = \delta_{ij}^2 \cdot \frac{2(x_{ij}^2 - m_{ij})}{\sigma_{ij}^2}. \tag{8}$$

In Eq. (8), $\delta_{ij}^2 (i = 1, \ldots, n \text{ and } j = 1, \ldots, p)$ represents the error signal from the $i$th node of the first layer to the $j$th node of the second layer, $x_{ij}^2$, ($i = 1, \ldots, n$ and $j = 1, \ldots, p$) is the input value from the $i$th node of the first layer to the $j$th node of the second layer, $\delta_{ij}^2 = \delta_j^3 \cdot \prod_{i=1}^{n} o_{ij}^2$, and $o_{ij}^2 = \exp[-(x_i^2 - m_{ij}/\sigma_{ij})^2]$. The regulated input layer rule can be derived as

$$m_{ij}(t+1) = m_{ij}(t) + \eta_m \cdot \delta_{ij}^2 \cdot \frac{2(x_{ij}^2 \cdot m_{ij})}{\sigma_{ij}^2}, \tag{9}$$

$$\sigma_{ij}(t+1) = \sigma_{ij}(t) + \eta_\sigma \cdot \delta_{ij}^2 \cdot \frac{2(x_{ij}^2 - m_{ij})^2}{\sigma_{ij}^3}. \tag{10}$$

In Eqs. (9) and (10) $\eta_m$ and $\eta_\sigma$ are the learning rates of the adjusted parameters ($m_{ij}$ and $\sigma_{ij}$), respectively.

Based on the above, the error function will be able to reach the minimum value. If a set of training data is input, the FNN control system will obtain an appropriate recipe (controllable variable). The direct proportional relationship that exists between the regulated range and the sensitivity of the error function weight in this control system implements the control task. The proposed control procedure is as follows:

- Step 1: The initial controller response will assume the nominal target ($T$). The actual response for the current period and expected response to the initial FNN controller are input. The controllable variable ($\hat{\mu}_t$) for the current period is predicted via the difference between the two responses.
- Step 2: The predicted controllable variable ($\hat{\mu}_t$) is substituted into the process model so that the actual response for the next period ($y_{t+1}$) can be obtained using measurement.
- Step 3: The error value ($e = T - y_{t+1}$) between the nominal target and actual response is computed.
- Step 4: The error value is transferred between the controllable variable for the current period and the predicted controllable variable ($(\mu_t - \hat{\mu}_t) = e/\beta$) to correct the FNN weight for the next run.